REMARKS

This Amendment is being filed in response to the Office Action mailed from the U.S. Patent and Trademark Office on May 10, 2007 in the above-identified application. Reconsideration and further examination are requested.

Claims 1-21 are pending in the application. Claims 1-4 and 6-21 have been rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,789,116 to Sarkissian et al. ("Sarkissian"). Claim 5 has been rejected under 35 U.S.C. 103 as being obvious over Sarkissian in view of U.S. Patent Application no. 2002/0052749A1 to Bauer ("Bauer").

Claims 1-5, and 21 have been amended. Claims 22-35 have been added. The amendments and new claims find support on Page 22, line 25 through Page 33, line 4 of the Specification and in FIG. 5, and do not introduce any new matter. Claims 6-20 have been cancelled without prejudice and may be pursued in a later filed application.

Claims 1 and 21 are independent claims upon which all others depend. Claim 1 recites:

> 1. A data engine located in a programmable pipeline processor for processing non-field delineated, streaming, application level database records received from a mass storage device, the data engine comprising:
>
> a data parser configured to parse non-field delineated database records received from the mass storage device into the field-delineated data;
>
> filter logic configured to receive field delineated database records from the database parser and to filter the field delineated data by performing a field operation on the field delineated database records; and
>
> an output tuple generator, configured to assemble filtered field delineated database records into an output tuple.

Claim 21 is an analogous method claim.

Without limitation, embodiments of the present invention relate to a data engine of a Programmable Streaming Data Processor (PSDP) which is arranged to perform primitive functions directly on database records. The data engine processes non-field delineated database records from a mass storage device, such as a disk drive, prior to its being

forwarded to a central processing unit (CPU) of a more general processor. The data engine allows the PSDP to perform certain preliminary processing in order to reduce the computational load on the local CPU.

As recited in independent claim 1, the data engine includes a data parser that determines or parses field boundaries in the non-delineated database records received from a streaming interface FIFO, selects one or more fields to be output tuples. Filter logic within the data engine determines whether an output tuple is to be selected for further processing by additional Job Processing Units, and an output generator assembles filtered field delineated database records into an output tuple.

Generally, the data engine receives non-field delineated data stream, and employs logical arithmetic methods to compare fields with one another, or with values otherwise supplied by general purpose processors to precisely determine which records are worth transferring to memory of the more general purpose distributed processors. The architecture allows for the use of substitution tables, temporary registers, and a data string register to assist in the efficiency and accuracy of the data engine processing.

The Claims of the Present Invention Are Not Anticipated By Sarkissian

Sarkissian describes a network packet monitor that may be employed on packets "passing through the connection point on a computer network." Sarkissian, Col. 3, lines 30-33. The packet monitor of Sarkissian "may be used to analyze traffic in a network." Sarkissian, Col 25 lines 24-25. The monitor comprises a parser sub-system that determines flow signatures and an analyzer sub-system that analyzes the flow signature of each packet. Sarkissian, Col 25 lines 24-25. In contrast to merely analyzing and routing packet flows in a network, the present invention relates to non-field delineated streaming application level database records, and provides processing and filtering on the non-field delineated database records received.

With respect to claim 1, the Examiner cited the parser interface 1101 of Fig. 11 in Sarkissian as the "interface, for receiving field-delineated data from a field parser." The "parser" of Sarkissian "examines the packets using pattern recognition process 204 that parses the packet and determines the protocol types and associated headers for each

protocol layer that exists in the packet 302 to match packets to data flows. Sarkissian, Col. 7, lines 63-67, and Col. 8, lines 64-67. In contrast, the data referred to in the present invention relates primarily to field-oriented database records, including field delineated data such as tables, indices, and views. For example, the system of the present invention can also be advantageously used to process less structured data, such as, variable length character strings, Binary Large Objects (BLOBS), graphics files, and the like. Specification, Page 18, lines 5-14.

The identification of protocol types and headers as performed in Sarkissian is much different than the application level, field-delineated data as described in connection with the present invention. This difference is of particular significance with regards to the "filter logic configured to receive field delineated database records from the database parser and to filter the field delineated data by performing a field operation on the field delineated database records." In other words, Sarkissian is processing only packet headers, and passes, unaltered, application level data, (i.e., the payload portion of his packet Sarkissian, Col. 6, lines 44-45 whereas the data engine presently claimed processes the database records in the payloads.

In addition, the "hash code" in Sarkissian is merely being used to determine if a new packet belongs to an existing flow via a database (table/lookup). That is not the same as parsing application level database records. The analyzer sub-system 303 of Sarkissian cited by the Examiner does not "***filter the field delineated data by performing a field operation on the field delineated database records.***" As described above, the data engine of the present claim parses field boundaries in non-field delineated database records and then performs low level logical operations to reduce computational load on a general processor. The processor of the present invention employs logical arithmetic methods to compare fields with one another, or with values otherwise supplied by general purpose processors to precisely determine which records are worth transferring to memory of the more general purpose distributed processors. The architecture allows for the use of substitution tables, temporary registers, and a data string register to assist in the efficiency and accuracy of the data engine processing.

The analyzer of Sarkissian does not actually perform a field operation on field delineated database records. The analyzer is simply a classification mechanism for analyzing header fields in the record and switching packet flows from a network. Therefore, because neither Sarkissian nor Bauer, taken alone or in combination makes obvious the field operation on the field delineated database records of Claims 1 or 21, Applicant respectfully submits that Claims 1 and 21 are in condition for allowance. Further, Applicant respectfully submits that Claims 2-5 and 22-35 are also in condition for allowance as dependent on an allowable base claim.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned.

Respectfully submitted,

HAMILTON, BROOK, SMITH &
REYNOLDS, P.C.

By _____
F. James Coe
Registration No. 46,630
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133
Date: October 19, 2007